CHAPTER 1

Overview of Deploying IIS 6.0

Deploying Internet Information Services (IIS) 6.0 identifies the key design and deployment processes that you must complete for your IIS 6.0 solution. This book provides prescriptive, task-based, and scenario-based guidance to help you design an IIS 6.0 solution and then deploy that solution within your organization. After you deploy and secure your IIS 6.0 solution as a platform for your Web applications, you can then deploy your Web applications. However, because the target audience of this book is Web server administrators, modifying Web applications is considered a developer-related topic and is not covered in this book.

In This Chapter

Overview of Deploying an IIS 6.0 Web Server

Organizations and individuals use Web sites and applications every day as a way to do business on the Internet and within their intranets. Internet Information Services (IIS) 6.0 helps you meet your business needs by providing the services to support a secure, available, and scalable Web server on which to run these Web sites and applications.

This chapter describes the high-level processes that are presented in this book for deploying a new IIS 6.0 Web server in your organization's production environment. The other chapters in this book are divided into separate IIS deployment topics that target a specific area of the deployment process including server security, application availability, deploying ASP.NET applications, Web site migration, and server upgrades. For a comprehensive understanding of IIS 6.0 deployment, read all of the chapters in sequential order. For information about a specific aspect of IIS 6.0 deployment, read the individual chapter that corresponds to your area of interest.

Everyone deploying IIS 6.0 needs to decide in which application isolation mode IIS should run. This book highlights worker process isolation mode because of the security and availability improvements from earlier versions of IIS. This book also compares worker process isolation mode to IIS 5.0 isolation mode, which is provided for maximum backward compatibility with existing applications. If your existing Web applications do not possess these characteristics, you should run IIS in worker process isolation mode.

Finally, while you prepare to deploy IIS 6.0, you must verify that your existing Web sites and applications are compatible with IIS 6.0 and with the Microsoft® Windows® Server 2003, Standard Edition; Windows® Server 2003, Enterprise Edition; Windows® Server 2003, Datacenter Edition; and Windows® Server 2003, Web Edition operating systems. Verification of the Web site and application compatibility should be done on a test Web server before deploying on a production Web server.

The processes in this book have been carefully developed and tested to provide a blueprint for an easy and comprehensive deployment of IIS 6.0. Following the recommendations presented during the deployment process will help your Web servers be as secure as possible and highly available.

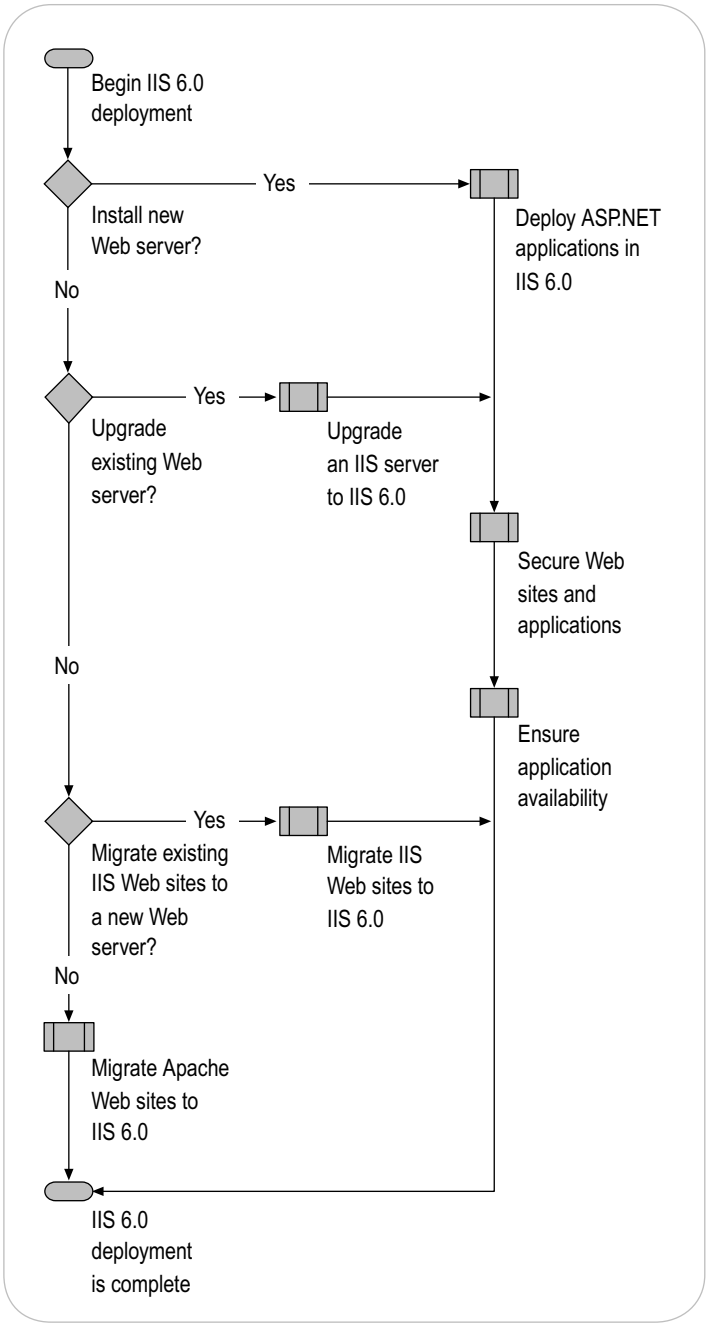
Process for Deploying an IIS 6.0 Web Server

The IIS 6.0 deployment process is written for Web server administrators who are responsible for installing and configuring IIS on new or existing servers. The chapters in this book can be divided into two main deployment scenarios:

- Deploying a new Web server running Windows Server 2003 with IIS 6.0
- Upgrading or migrating to a Web server running Windows Server 2003 with IIS 6.0

Figure 1.1 illustrates the chapters in this book that correspond to these scenarios. Review the flowchart in this figure and read the descriptions of each chapter to help identify the deployment tasks that you need to complete for your IIS 6.0 solution and to discover which chapters will help you complete those tasks.

Figure 1.1 IIS 6.0 Deployment Process Steps and the Corresponding Chapters



Each chapter includes a flowchart that represents the organizing principle of the chapter content. Each chapter also includes a quick-start guide to facilitate the fastest possible deployment. You can use these quick-start guides to help identify the steps of the deployment process that you need additional information to complete, and then you can skip the information with which you are already familiar.

Deploying a New IIS 6.0 Web Server

The following chapters in this book are about deploying a new server running IIS 6.0 to host your Web sites and applications:

- Chapter 2: "Deploying ASP.NET Applications in IIS 6.0"

  Read this chapter to understand specific considerations for deploying ASP.NET applications in IIS 6.0. In particular, the chapter describes how you can run multiple versions of the Microsoft .NET Framework on the same Web server and how you can configure ASP.NET applications to use the appropriate version of the .NET Framework.

  After completing the process in this chapter, you must still optimize your server for security and scalability. The remaining chapters in this book focus on topics that are applicable to all Web applications, including ASP.NET applications, noting exceptions where appropriate.

> **Important**
> ASP.NET is not available on the following operating systems: Microsoft® Windows® XP 64-Bit Edition; the 64-bit version of Windows® Server 2003, Enterprise Edition; and the 64-bit version of Windows® Server 2003, Datacenter Edition. For more information, see "Features unavailable on 64-bit versions of the Windows Server 2003 family" in Help and Support Center for Windows Server 2003.

- Chapter 3: "Securing Web Sites and Applications"

  Read this chapter to learn how you can further protect the Web sites and applications that are hosted on a Web server running IIS 6.0. This chapter describes how to secure a Web server and how to secure individual Web sites and applications running on IIS 6.0. In particular, this chapter describes the process for reducing the attack surface of a Web server, preventing unauthorized access to Web sites and applications, isolating Web sites and applications, configuring user authentication, encrypting confidential data exchanged with clients, and maintaining the security of Web sites and applications.

- Chapter 4: “Ensuring Application Availability”

  Read this chapter to learn how you can fully utilize the features of IIS 6.0 to enhance the availability of your Web applications. This chapter describes how you can set realistic application availability goals and then verify that you are attaining those goals. In addition, it describes how you can configure the features of worker process isolation mode for optimum application availability, depending on your business needs and the processing load on your applications. Finally, this chapter identifies common application incompatibilities with IIS 6.0, and it explains how you can test your applications for compatibility with worker process isolation mode.

Upgrading and Migrating a Server to IIS 6.0

The following chapters in this book assume that you have Web sites and applications that are hosted on an existing Web server running Apache or an earlier version of IIS. Select one of the following chapters to perform the deployment of IIS 6.0 when you have an existing server:

- Chapter 5: “Upgrading an IIS Server to IIS 6.0”

  Read this chapter when you are upgrading Web sites and applications that are hosted on a server running the Microsoft Windows NT® Server 4.0 operating system with IIS 4.0 or a server running the Microsoft® Windows® 2000 Server operating system with IIS 5.0, to a server running Windows Server 2003 with IIS 6.0. This chapter defines *upgrading* as the process of updating a Web server running IIS 4.0 or IIS 5.0 to run IIS 6.0 when the Web sites and applications hosted by that Web server are not moved to another server.

- Chapter 6: “Migrating IIS Web Sites to IIS 6.0”

  Read this chapter when you are migrating Web sites that are hosted on a server running Windows NT Server 4.0 with IIS 4.0, or a server running Windows 2000 Server with IIS 5.0, to a newly installed server running Windows Server 2003 with IIS 6.0. This chapter assumes that you are moving Web sites from an existing Web server running an earlier version of IIS, to a newly installed Web server running IIS 6.0. This chapter explains how you can use the IIS 6.0 Migration Tool to migrate your IIS Web sites to IIS 6.0. This chapter also describes a manual process for migrating IIS Web sites to IIS 6.0.

- Chapter 7: “Migrating Apache Web Sites to IIS 6.0”

  Read this chapter when you are migrating Apache Web sites that are hosted on an existing server to a server running IIS 6.0 and Windows Server 2003. This chapter assumes that you are moving applications from an existing Web server running a Linux-based operating system with Apache to a newly installed Web server running Windows Server 2003 with IIS 6.0. The process in this chapter describes how you can use the Apache to IIS 6.0 Migration Tool to migrate your Apache Web sites to IIS 6.0.

Overview of IIS 6.0

IIS 6.0 with Windows Server 2003 provides integrated, reliable, scalable, secure, and manageable Web server capabilities over an intranet or the Internet. IIS is a stable and secure platform for running dynamic network applications. Organizations of all sizes use IIS to host and manage Web sites on the Internet or on their intranets, to host and manage FTP sites, and to route news or mail by using the Network News Transfer Protocol (NNTP) and the Simple Mail Transfer Protocol (SMTP).

IIS 6.0 takes advantage of the latest Web standards like ASP.NET, XML, and Simple Object Access Protocol (SOAP) for the development, implementation, and management of Web applications. IIS 6.0 includes new features that are designed to help organizations, IT professionals, and Web administrators achieve their goals of performance, reliability, scalability, and security for potentially thousands of Web sites hosted on a single server or on multiple servers.

IIS 6.0 Benefits and Features

IIS 6.0 provides the following benefits and features:

- **Reliability.** IIS 6.0 uses a new request-processing architecture and application isolation environment that enables individual Web applications to function within a self-contained worker process. This environment prevents one application or Web site from stopping another, and it reduces the amount of time that administrators spend restarting services to correct application-related problems. The new environment also includes proactive health monitoring for application pools. For more information about application reliability in IIS 6.0, see “Ensuring Application Availability” in this book.
- **Scalability.** IIS 6.0 introduces a new kernel-mode driver for Hypertext Transfer Protocol (HTTP) parsing and caching that is specifically tuned to increase Web server throughput and scalability of multiprocessor computers. The result is an increase in the following:
  - The number of Web sites that a single IIS 6.0 server can host
  - The number of concurrently active worker processes
  - The performance for startup and shutdown times for the Web server and for individual Web sites
  - The number of simultaneous requests that a Web server can service.

Also, by configuring the startup and shutdown time limits for worker processes, IIS allocates resources to active Web sites instead of keeping resources on idle requests.

- **Security.** IIS 6.0 provides significantly improved security over IIS 5.0. For example, to reduce the attack surface of systems, IIS 6.0 is not installed by default on Windows Server 2003, Standard Edition; Windows Server 2003, Enterprise Edition; and Windows Server 2003, Datacenter Edition. After installing these products, administrators must manually install IIS 6.0. When IIS 6.0 is installed, it is locked down by default so that it can serve only static content. By using the Web Service Extensions node in IIS Manager, Web site administrators can enable or disable IIS functionality based on the individual needs of their organization.

  IIS 6.0 includes a variety of security features and technologies to help ensure the integrity of your Web and FTP site content, as well as the data that is transmitted through your sites. These security features and technologies include Advanced Digest authentication, improved access control, Secure Sockets Layer (SSL) encryption, centralized certificate storage, and detailed auditing capabilities.

  For more information about IIS security, see "Securing Web Sites and Applications" in this book; and see "Security" in IIS 6.0 Help, which is accessible from IIS Manager.

- **Manageability.** To meet the needs of a diverse set of organizations, IIS 6.0 provides a variety of manageability and administration tools. Administrators can configure an IIS 6.0 server by using IIS Manager, by running administration scripts, or by directly editing the IIS metabase. Administrators can also remotely administer IIS servers and Web sites.

- **Enhanced Development.** Compared to Windows 2000 Server, Windows Server 2003 offers an improved developer experience with ASP.NET and IIS integration. ASP.NET runs most Active Server Pages (ASP) code while providing greater functionality for building enterprise-class Web applications that can work as a part of the .NET Framework. Use ASP.NET to fully utilize the features of the common language runtime, such as type safety, inheritance, language interoperability, and versioning. IIS 6.0 also offers support for the latest Web standards including XML, SOAP, and Internet Protocol version 6 (IPv6).

  For more information about enhanced development of ASP.NET applications, see "Deploying ASP.NET Applications in IIS 6.0" in this book.

- **Application Compatibility.** According to feedback from thousands of customers and independent software vendors (ISVs), IIS 6.0 is compatible with most of their existing Web applications. Also, to ensure maximum compatibility, you can configure IIS 6.0 to run in IIS 5.0 isolation mode.

Internet and Intranet Applications on IIS 6.0

The IIS 6.0 deployment process can be applied to Web sites and applications that are hosted on the Internet or within the intranet of your organization. Throughout the deployment process, explicit references are made for deployment considerations relating to Web servers facing the Internet or within an intranet.

Note

The deployment processes and recommendations described in this book can be used to deploy an Internet or intranet Web server, unless otherwise noted.

Determining Application Compatibility with IIS 6.0

One of the primary concerns when deploying IIS 6.0 is whether or not your existing applications are compatible with IIS 6.0. Windows Server 2003 and IIS 6.0 are designed to provide maximum application compatibility. In most cases, existing Web sites and applications will run on IIS 6.0 without modification.

The deployment process in this book provides prescriptive guidance about how to address known Web site and application compatibility issues. For a more detailed explanation of Web site and application compatibility with IIS 6.0, read the following:

- "Preparing for Upgrade" in "Upgrading an IIS Server to IIS 6.0" in this book when you are performing an upgrade of an existing Web server running an earlier version of IIS.
- "Preparing for Migration" in "Migrating IIS Web Sites to IIS 6.0" in this book when you are moving existing Web sites and applications that are hosted on a Web server running an earlier version of IIS to a newly installed Web server running IIS 6.0.
- "Preparing for Migration" in "Migrating Apache Web Sites to IIS 6.0" in this book when you are moving existing Web sites and applications that are hosted on a Web server running Apache to a newly installed Web server running IIS 6.0.

For a more detailed explanation of application compatibility with Windows Server 2003, see "Planning and Testing for Application Deployment" in *Planning, Testing, and Piloting Deployment Projects* of this kit.

Moving from IIS 5.0 Isolation Mode to Worker Process Isolation Mode

IIS 6.0 can run in one of two distinct modes of operation, which are called application isolation modes. *Application isolation* is the separation of applications by process boundaries that prevent the applications from affecting one another, and it is configured differently for each of the two IIS application isolation modes: IIS 5.0 isolation mode and worker process isolation mode.

Before you begin deployment, review the following:

- Differences between IIS 5.0 isolation mode and worker process isolation mode
- Benefits of moving from IIS 5.0 isolation mode to worker process isolation mode

Note

This book assumes that IIS 6.0 is running in worker process isolation mode, unless otherwise noted.

Reviewing Application Isolation Modes

Worker process isolation mode uses the redesigned architecture for IIS 6.0. This isolation mode runs all application code in an isolated environment. However, unlike earlier versions of IIS, IIS 6.0 provides isolation without a performance penalty because fewer processor instructions are ran when switching from one application pool to another. Worker process isolation mode is compatible with most existing Web sites and applications. Whenever possible, run IIS 6.0 in worker process isolation mode to benefit from the enhanced performance and security in IIS 6.0.

IIS 5.0 isolation mode provides compatibility for applications that depend upon the process behavior and memory model of IIS 5.0. Run IIS in this mode only when a Web site or application cannot run in worker process isolation mode, and run it only until the compatibility issues are resolved.

Important

IIS 6.0 cannot run both application isolation modes simultaneously on the same server. Therefore, on a single server running IIS 6.0, you cannot run some Web applications in worker process isolation mode and others in IIS 5.0 isolation mode. If you have applications that require separate modes, you must run them on separate servers.

IIS 6.0 defaults to a different application isolation mode based on the type of deployment you select. For new installations and migrations, IIS is configured to run in worker process isolation mode by default. After you perform an upgrade from an earlier version of IIS, IIS is configured to run in IIS 5.0 isolation mode by default.

Before configuring IIS 6.0 to run in worker process isolation mode, evaluate whether your Web sites and applications are compatible with worker process isolation mode. In most cases, IIS hosts your Web sites and applications in worker process isolation mode without any problems. Nevertheless, determine application compatibility in your lab before deploying your IIS solution into production.

For more information about worker process isolation mode, IIS 5.0 isolation mode, and evaluating Web site and application compatibility with worker process isolation mode, see "Determining Application Compatibility with Worker Process Isolation Mode" in "Upgrading an IIS Server to IIS 6.0" in this book.

Note

Identifying a complete list of potential incompatibilities that applications can experience with worker process isolation mode is beyond the scope of this book. Even after following the guidelines in this chapter, you still need to verify in your lab whether your Web sites and applications are compatible with worker process isolation mode.

Benefits of Moving to Worker Process Isolation Mode

Worker process isolation mode provides higher levels of security and availability for Web sites and applications than IIS 5.0 isolation mode. Therefore, it is recommended that you configure IIS 6.0 to run in worker process isolation mode.

In IIS 5.0, applications can be pooled together out-of-process, but in only one application pool. In IIS 6.0, worker process isolation mode supports multiple application pools, where each application pool can have a different configuration, such as a unique recycling configuration. Therefore, you can prevent a single Web site or application that periodically fails from disrupting other Web sites and applications. In addition, worker process isolation mode provides the following improvements to IIS.

Security Enhancements

IIS 6.0 includes a variety of security features and technologies that help ensure the integrity of your Web site content, and of the data that is transmitted through your sites. The following security enhancement is only available when IIS 6.0 is running in worker process isolation mode.

Default process identity for Web sites and applications set to NetworkService

In IIS 5.0 isolation mode, the default process identity is LocalSystem, which enables access to, and the ability to alter, nearly all of the resources on the Web server.

Performance and Scaling Enhancements

Future growth in the utilization of your Web sites and applications requires increased performance and scalability of Web servers. By increasing the speed at which HTTP requests can be processed and by allowing more applications and sites to run on one Web server, the number of Web servers that you need to host a site is reduced. The following are a few of the performance improvements included in worker process isolation mode.

Support for processor affinity for worker processes in an application pool

You can configure all of the worker processes in an application pool to have affinity with specific processors in a multiprocessor or server. Processor affinity allows the worker processes to take advantage of more frequent processor caching (Level 1 or Level 2).

Elimination of inactive worker processes and reclamation of unused resources

You can configure application pools to have worker processes request a shutdown if they are idle for a certain amount of time. This can free unused resources for other active worker processes. New worker processes are then started only when they are needed.

Distributing client connections across multiple worker processes

You can configure an application pool to have more than one worker process servicing client connections, also known as a *Web garden*. Because there are multiple worker processes, the incoming client connections are distributed across the worker processes and throughput is not constrained by a single worker process.

Ability to isolate Web sites and applications from each other

You can isolate Web sites and applications without incurring a performance penalty.

Availability Enhancements

Because worker process boundaries isolate the applications in an application pool from the applications in other application pools, if an application fails, it does not affect the availability of other applications running on the server. Deploying applications in application pools is a primary advantage of running IIS 6.0 in worker process isolation mode.

Reduced number of server restarts that are required when administering Web sites and applications

Many of the common operation tasks do not force the restart of the server or the Web service. These tasks, such as upgrading site content or components, debugging Web applications, or dealing with faulty Web applications, can be performed without affecting service to other Web sites or applications on the server.

A fault-tolerant request-processing model for Web sites and applications

In IIS 5.0 isolation mode, each Web site or application has only one worker process. However, in worker process isolation mode, you can create a *Web garden* by configuring a number of worker processes to share the processing. The benefit of a Web garden is that if one worker process stops responding, other worker processes are available to accept and process requests.

Isolation of failed worker processes from healthy worker processes

In worker process isolation mode, IIS can determine that a worker process has failed and start a new worker process. To minimize the interruption of service, new requests are queued until the new worker process is active. In the case where a worker process has not yet failed but is considered unhealthy, IIS starts a new worker process. When the new worker process is active, IIS shuts down the unhealthy worker process. After IIS creates the new worker process, the failed worker process can be separated, or *orphaned*, from the application pool. The advantage of orphaning a worker process rather than terminating it is that debugging can be performed on the orphaned worker process.

Health monitoring of Web sites and applications

In worker process isolation mode, you can configure an application pool to monitor not only the health of the entire application pool, but also individual worker processes servicing the application pool. Monitoring the health of a worker process allows IIS to detect that a worker process is unable to serve requests and to take corrective action, such as recycling the failed worker process.

In addition, worker process isolation supports other responses when a failed worker process or application pool is detected. For example, IIS can attach a debugger to an orphaned worker process or notify an administrator that an application pool has failed due to rapid-fail protection.

Prevention of Web sites or applications that fail quickly from consuming system resources

In some instances, availability can be affected by Web sites and applications that fail very quickly, are automatically restarted, and then fail quickly again. The endless cycle of failure and restarting can consume system resources, causing other Web sites and applications to experience denial of services because of system resource shortages.

Worker process isolation mode includes *rapid-fail protection* that stops an application pool when too many of the worker processes assigned to an application pool are found to be unhealthy within a specified period of time.

Automatic restart of poorly performing Web sites and applications

Some Web sites and applications have memory leaks, are poorly coded, or have other unidentified problems. In IIS 5.0 isolation mode, these applications can force you to restart the entire Web server. The recycling feature in worker process isolation mode can periodically restart the worker processes in an application pool without affecting service availability. Worker processes can be scheduled to restart based on several options, such as elapsed time or the number of requests served.